DOCUMENT RESUME

ED 231 145 EC 152 522

AUTHOR Fuchs, Victor E.
TITLE Software Marketing Considerations.
PUB DATE Mar 83
NOTE 12p.; Paper presented at the Council for Exceptional Children National Conference on the Use of Microcomputers in Special Education (Hartford, CT, March 10-12, 1983).
PUB TYPE Viewpoints (120)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Computer Programs; Educational Change; Educational Technology; Elementary Secondary Education; Futures (of Society); Higher Education; Industry; *Instructional Materials; *Marketing; Purchasing; *School Business Relationship

ABSTRACT

Seven factors that currently affect the potential for marketing and publishing computer software for education are discussed: (1) computers as an inplace technology in education, (2) marketing and distribution patterns for software, (3) consumer demand, (4) quality, (5) timelessenss, (6) basic skills, and (7) the future. The proliferation of computers in the schools creates a growing computer base to utilize whatever software is produced. As the industry learns how to effectively market and distribute software, the distribution of software products will be as widespread and well used as the distribution of print or audiovisual products. Consumer demand for software is affected by three constituents: the software publisher/producer/distributor, the purchaser, and the user. School acquisition practices are influencing consumer demand, since publishers seek areas that they can adapt software to match nonsoftware products they currently produce and distribute. Consumer demand for software ultimately will influence software development greatly. It is important to remember that quality software sells, and software that will retain its educational usefulness and relevancy over long periods will be needed by schools. In addition, software/courseware that helps students achieve minimum competency standards--the basic skills--will be attractive. Computer technology in education will force changes in the delivery and methods by which teaching and education are practiced. (SEW)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☑ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

ED231145

Software Marketing Considerations

by

Victor E. Fuchs

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Victor E Fuchs

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

LINC Resources, Inc.
1875 Morse Road
Columbus, Ohio 43229

EC152202

Software Marketing Considerations

by Victor E. Fuchs

The increasing use of technology, especially computer technology, by educators, requires us to learn from the experience of others so that we do not paint ourselves into a corner. Advanced computer technology is going to change the way we educate our students and the way we relate and work with students. It will require us to change, to learn new skills, and to manage our skills differently from the way we have been taught--different from the comfortable ways of the past.

We can learn part of this from the experience of the American automobile industry. The American automobile industry is recovering from its worst decline in its history because it did not respond to the need for change early enough. The decline was brought about by the industry itself due to a confluence of factors it could not control, but did not want to fully recognize and deal with at a time when everything else seemed to be all right. Information available to industry analysts in the late '60s and early '70s profiled the factors: rising oil and steel prices, management/labor practices needing revision, increasing foreign automobile sales, and developing government regulations requiring production line changes--changes partly inspired by products available from foreign manufacturers. Despite all of this information, the U.S. automobile industry resisted change and did not make the significant decisions required before the sales started to plummet. It was a classic pattern of an industry: a set of people not willing to change and make the investment required. It was easier to take the profits than the tax credits. It was productivity now at the expense of long-term, stable productivity in the future. It was management and the labor force that wanted continuing productivity and salary increases without adjusting to the

advances in technology produced by others. The results, as we know, have been disastrous. Thousands of automobile workers are unemployed, small business firms supplying and relying on the major automobile manufacturers have been put out of business, others are hanging on by a thread, and the resultant loss of tax revenue combined with increased expenditures for unemployment compensation are dramatically increasing the federal deficit.

The automobile industry does not stand alone; it has reflected what a lot of us do by practice. We resist change, under the guise of adapting to it, when we know that more fundamental modification to our current practice is needed to adequately meet the challenges of the future. The electronic revolution, just beginning, is going to necessitate fundamental changes in our educational practice and thinking.

As educators, we need to look at the signs around us--signs that to varying degrees parallel those that have been predominant in the auto industry. The basic elements that are required to make our education system function are increasing in cost; our management/labor practices need revision; foreign competitors, in the form of the electronics industry, are beginning to produce technology products that will compete with our traditional practice of producing teachers that convey information to students; and government regulations continue to shape the way in which we go about the business of education. It's time we asked ourselves a serious question. Is the quality of the educational product we are producing today good enough for the users and buyers of that product tomorrow, and can we produce a better product in less time, with less cost? Computer technology will force us to do it.

The changes we must initiate will reshape and create our roles as educators directing technology, rather than educators searching for ways to make technology useful. One of the most practical ways we can do this is to

keep a tight grasp on content and how we communicate content to students, whether handicapped or nonhandicapped. Computer technology is technology, nothing more or less. What we do with it and how we use it is important, but what we put into it, how we influence and shape the content of the computer program--the software--is initially important. Otherwise, we have the proverbial garbage in, garbage out. The medium is not the message unless we allow it to be. The medium of the computer should not be more important than the quality of the computer program. Unfortunately, producing quality has never been easy. Quality--to use a phrase--is job one.

We at LINC have been dealing for some time with the medium and the message issue in the various formats of print, audiovisuals, telecommunications, computer software and advanced electronic technology. We have also dealt with the processes required to arrange for the commercial production and distribution of innovative products that utilize different formats. In the process, we have learned something about the factors that make a product marketable, regardless of its format, whether it's a book, a computer program or an electronic device. In all of this, the single most important factor is not the medium, but the message, and how easily it can be used and communicated to others. Computer software is not an exception. Let's look at the market factors that surround computer software and what will influence its marketability and publishability.

Factor #1: Computers as an Inplace Technology

There is no doubt but that computers have been around a long time. It is almost an embarrassment to see us as educators making such a fuss over them, since they are not a new technology. What has happened that is exciting with computers is miniaturization--the ability to have your own personal computer. The distinctions between microcomputers, very small computers,

minicomputers, small computers, computers not so small, and mainframes--the big ones--are diminishing. Today it is possible to have a mainframe computer on your desk top, ready for personal use. The HP-9000 is an example; it will compute one million instructions a second.

As of December 1982, approximately 9,350 school districts had microcomputers for classroom use. About 25,000 school buildings had micros, and the relationship between the size of the student enrollment and the number of computers required by the school was proportional. Increasing numbers of teachers are becoming computer literate; many are beginning to write their own software and courseware. It is projected that by the end of 1983 the number of schools to have acquired computers will have doubled. By 1985 most schools will have an installed computer base; by 1987 only a few schools will not have a computer.

The rate at which the number of computers in the schools is growing is slow compared to the fast pace of computer software and courseware development. Presently there are some 700 plus publishing and software houses either developing, producing, or distributing software. This does not take into account the number of educators producing software or the number of students and parents producing software. The importance of this marketing factor--the proliferation of computers in the schools--is that there is a growing computer base to utilize whatever software is produced. Therefore, if you produce a software program, it is probable that it can be used by more and more educators and students.

Factor #2: Marketing and Distribution Patterns for Software

A preliminary analysis of a poll of producers and distributors of software by LINC in the late fall of 1982 showed the following marketing and distribution trends:

- 80 percent of software publishers still use direct mail supported by catalogs, fliers, ads, exhibits, etc.
- 50 percent of the companies have some kind of sales representative or dealer network. This is increasing, since one of the most effective ways to sell software is an onsite demonstration, and software sales are presently considered building level sales.
- An increasing number of local computer retailers (Computer Land, Radio Shack, Atari, Sears, etc.) are becoming involved in the school/home market.
- 20 percent of companies have a preview policy to protect from unauthorized copying of disks.
- Only a minority of companies give unrestricted previews.
- 15 percent are experimenting with the demo disk concept, sometimes charging for a demo disk.
- Companies using direct mail consider warranties and guarantees an alternative to a preview policy; however, 20 to 25 percent of companies do not have a specific return policy; you buy it, you own it. Industry standards are beginning to emerge.
- 75 percent of companies guarantee their diskettes will "boot up," and 50 percent offer warranties that the courseware/software will play/operate.
- Warranties range from ten days to two years, thirty days is common; ninety days to six months is offered by a number of companies; ninety days will probably become the standard.
- Companies are experimenting with backup and inexpensive replacement diskettes when the original disk needs to be replaced, updated, or revised.

The implication for this marketing factor is that as the industry learns how to effectively market and distribute software, the distribution of software products will be as widespread and well-used as the distribution of print or A-V products.

Factor #3: Consumer Demand for Software

Consumer demand for software is affected by three constituents: the software publisher/producer/distributor, the purchaser, and the user.

The software publisher/producer/distributor, as we stated before, views the market for software as a building level sale. Some believe that as states and large school districts consider the notion of software libraries, like state and district level film libraries, the volume of software sales will decrease while the price per copy will increase. This happened to the film industry. If this does occur, many software publishers will re-evaluate their production schedules and their marketing strategies. The demand for software under these conditions will create an occasional use market for software, limiting much software to the supplemental materials category. In response, publishers will market as they perceive the market through the acquisition practices of software libraries. Publishers will seek areas that they can adapt software to that match non-software products they currently produce and distribute. This is happening and will continue to happen. The result is the computerized workbook, the computer animated word list, the computer form of the programmed text. School acquisition practices and product line continuity are invisibly influencing consumer demand, creating an environment where the educator does not get to choose what corresponds to student abilities, but is presented with a predetermined selection of software that is influenced more by economics than instructional needs. The problem is that individual educators either do not have the opportunity to or cannot articulate what the

level of instructional quality is that they expect from a piece of software. This is not unique to software, but until individual educators make such a statement, they, as consumers of software, will have varying degrees of success in influencing and making known their demands.

On the other hand, purchasers of software are having the greatest amount of influence on consumer demand for software. Software acquisition decisions are primarily being made by media center personnel, curriculum coordinators and supervisors, department heads, library media specialists, and not enough teachers. At a recent NEA regional conference in the northeast, 50 percent of the software purchased was reported as not being used by the teachers. Reason: not enough teachers have a voice in the software purchase decision. The consumer with the most influence is one step away from the classroom where the real consumers, the students, exist. This is a situation that must be corrected if consumer demands for software are to be realistic.

Students at the present time have little influence on software demands and, therefore, on what type of software is produced, marketed and distributed. Change in this area is on the way for several reasons: first, when enough software is available on any specific topic so that students can choose among software options, it may be possible to match software to specific student learning styles and needs; second, student respect for teacher competence in using computers and software will inherently influence teacher selection of software that will be in response to student expectations of the instructor, therefore, students will affect software selection through this influence on teacher behavior; third, students will strive for computer literacy and better performance levels using computers and software to gain peer recognition, as well as teacher praise, that will produce a type of consumer pressure similar to that which students had on instructional

television; and last, the development of computer clubs as school student organizations will enable students to collectively influence software acquisition.

Consumer demand for software ultimately will influence software development to the maximum. Articulating the demand differences between administrators, specialists, teachers and students, will probably be, as it has in the past, more of a political process than an educational process. The relevance of consumer demand as a marketing consideration is to understand, when you develop your software, how it will affect the acceptance of the software. Caution: don't think that consumers won't buy a piece of junk. They will--if there is nothing else available.

Factors 4, 5, 6 and 7

Factor 4 is quality. Quality sells. Whatever you develop, do it right or don't do it at all. Educators are looking for quality courseware and software, and they're learning fast about the characteristics of poor quality in software. According to a recent article in the Wall Street Journal, software consumers are becoming more discriminating buyers. Given that all other things are equal, people will not choose to buy something of inferior quality, if for the same price, they can have something better. Caution: don't mistake junk for quality.

Factor 5, timelessness. Software that will retain its educational usefulness and relevancy over long periods of time will be sought after by budget-pinched schools. The ability to be used over and over again, without revision, is a key marketing factor. Schools do not want to be trapped into buying new software programs every year to replace the ones used the previous year. Schools do not want to look on software as consumables, but rather, as extensions of the computer hardware. As such, timelessness--the ability to

use the software for as long as you can use the hardware--will find a receptive market in the school. Caution: don't mistake software that uses student time to no end as timelessness. It's time-consuming junk.

Factor 6, basic skills. Software/courseware that helps students achieve minimum competency standards--the basic skills--will be attractive. Period. Caution: don't think that software that teaches the obvious is basic skills. It's teaching the student what they already know. Give an A to the student though the software is junk.

Factor 7, the future. The software of today will be the forgotten ware of tomorrow. Computers use software like lungs use air: a constant inhale and exhale, constantly using more, consuming what it captures, and disposing of what is not needed. Advances in computer technology will dwarf the ability of the individual to singlehandedly develop software. Most software will be developed by software companies and eventually software as individual diskettes or pieces of courseware will not exist, except for highly specific applications. "Information ware," time shared data banks containing the total of man's knowledge and information, will be available on a circulation, usage or computer readership basis. You'll get anything you want, but only pay for what you need. Computer programs will automatically delete all superfluous information not relevant to a user's needs or request. Portable operating systems will allow students to "plug in" to any computer terminal within transmitting distance wherever they are: increased storage of digitized information will allow you to carry around and access an encyclopedia; keyboards will be obsolete; commands to the computer will give way to interactive conversations. Ultimately, no one will write computer programs or software, as no one today "writes" language. The emphasis will be on managing the data and the environment it is presented in, specifying the

interrelationships between sets of information in order to protect objectivity, so that users will be able to draw their own conclusions from the data and meet their own needs by being guided into understanding information that can be useful to their lives.

In the final analysis, the concerns about how we use and make computer software in the future will be the concerns we have today. The marketing considerations for the future is to concentrate on what you know how to do now, to communicate your skills, your knowledge, your message, in a way that someone else can use it. This is the education process; it is the essence of what makes any piece of software marketable--the unique, creative, intellectual way you develop it so that it is useful to someone else. Succinctly put, make your software an original; don't copy someone else's. Computers can do that for you. It is the creator of the software who thinks, not the computer that runs it. You, the author, are the greatest asset of the software, now and in the future. Caution: don't abdicate the control of the content of the software programs to someone who doesn't know the subject. To do so will result in an inferior software program that will greatly reduce its marketability/publishability, but if you don't know how to design a computer program or to program it, get help from computer professionals. They know how to program better than you do. Practice on your own, not on consumers.